

```

using System;

using Google.Api.Gax.ResourceNames;

using Google.Cloud.Dlp.V2;

using static Google.Cloud.Dlp.V2.CustomInfoType.Types;

public class InspectStringCustomOmitOverlap
{
    public static InspectContentResponse Inspect(string projectId, string textToInspect)
    {
        var dlp = DlpServiceClient.Create();

        var byteItem = new ByteContentItem
        {
            Type = ByteContentItem.Types.BytesType.TextUtf8,
            Data = Google.Protobuf.ByteString.CopyFromUtf8(textToInspect)
        };

        var contentItem = new ContentItem { ByteItem = byteItem };

        var customInfoType = new CustomInfoType
        {
            InfoType = new InfoType { Name = "VIP_DETECTOR" },
            Regex = new CustomInfoType.Types.Regex { Pattern = "Larry Page|Sergey Brin" },
            ExclusionType = ExclusionType.Exclude
        };

        var exclusionRule = new ExclusionRule
        {
            ExcludeInfoTypes = new ExcludeInfoTypes { InfoTypes = { customInfoType.InfoType } },
            MatchingType = MatchingType.FullMatch
        }
    }
}

```

```
};
```

```
var ruleSet = new InspectionRuleSet  
{  
    InfoTypes = { new InfoType { Name = "PERSON_NAME" } },  
    Rules = { new InspectionRule { ExclusionRule = exclusionRule } }  
};
```

```
var config = new InspectConfig  
{  
    InfoTypes = { new InfoType { Name = "PERSON_NAME" } },  
    CustomInfoTypes = { customInfoType },  
    IncludeQuote = true,  
    RuleSet = { ruleSet }  
};
```

```
var request = new InspectContentRequest  
{  
    Parent = new LocationName(projectId, "global").ToString(),  
    Item = contentItem,  
    InspectConfig = config  
};
```

```
var response = dlp.InspectContent(request);
```

```
Console.WriteLine($"Findings: {response.Result.Findings.Count}");
```

```
foreach (var f in response.Result.Findings)
```

```
{  
    Console.WriteLine("\tQuote: " + f.Quote);  
    Console.WriteLine("\tInfo type: " + f.InfoType.Name);  
    Console.WriteLine("\tLikelihood: " + f.Likelihood);
```

```
}
```

```
return response;
```

```
}
```

```
}
```